Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Please **amend** claims 29-31 and 84.

1. (Previously Presented) A method for processing a database command, performed by an alternate database engine, the method comprising:

receiving, from a user, a database command requiring data from a first database engine, the first database engine having a command layer for processing database commands; and separately processing the database command using a command layer of an alternate database engine without accessing the command layer of the first database engine.

2. (Original) The method of claim 1, wherein the first database engine stores the data in a first database file.

3. (Original) The method of claim 1, wherein the alternate database engine stores second data in a second database file.

4. (Previously Presented) The method of claim 1, wherein the database command is compatible with one or more of: a Structured Query Language format, a Javascript Database Connectivity (JDBC) protocol and/or an Open-Database Connectivity protocol.

5. (Original) The method of claim 1, wherein the database command is a query.

6. (Original) The method of claim 5, said processing the database command further comprising:

evaluating the query.

7. (Previously Presented) The method of claim 6, said evaluating further comprising:

evaluating the query against system usage.

8. (Previously Presented) The method of claim 7, said evaluating further comprising:

evaluating the query based on one or more of: a parameter of the query, a number of relational databases to be accessed for the query, a size of a data field to be searched for the query, an availability of resources of a system maintaining the alternate database engine, an availability of resources of a system maintaining the first database engine, a number of relational database tables to be employed for the query, a limitation imposed on a size of a query result set, a number of columns of data to be returned in a query result set, a cost of a similar stored query and/or a number of function calls for the query.

9. (Original) The method of claim 7, further comprising:

submitting the query to the alternate database engine with a limit on a number of returns responsive to the query, based on said evaluating.

10. (Original) The method of claim 7, further comprising

editing the query, based on said evaluating.

11. (Original) The method of claim 7, further comprising:

rejecting the query, based on said evaluating.

12. (Previously Presented) The method of claim 6, wherein said evaluating comprises:

determining, prior to said processing, whether the database command requires accessing the first database engine, and if not, accessing data stored only by the alternate database engine.

13. (Original) The method of claim 12, said determining further comprising:

translating the query to a native format of the alternate database engine.

14. (Original) The method of claim 6, said evaluating further comprising:

determining whether the query requires accessing temporally sensitive data, and if so, accessing a transaction log of the first database engine.

15. (Original) The method of claim 5, further comprising:

generating a result of the query

16. (Original) The method of claim 15, further comprising:

transmitting the result to the one of the plurality of users submitting the database command.

17. (Original) The method of claim 16, wherein said transmitting further comprises:

transmitting the result in a format of the first database engine.

18. (Original) The method of claim 1, further comprising:

storing second data in a database file maintained by the alternate database engine.

19. (Original) The method of claim 18, said processing further comprising:

determining whether the database command requires at least a portion of said second data, and if so, identifying said portion responsive to the database command.

20. (Original) The method of claim 1, further comprising:

receiving new data to be provided responsive to database commands; and

storing said new data in a database file maintained by the alternate database engine.

21. (Original) The method of claim 1, further comprising:

receiving new data to be provided responsive to database commands; and

storing said new data in a database file maintained by the first database engine.

22. (Original) The method of claim 1, said processing further comprising:

translating the database command to a native format of the alternate database engine.

23. (Original) The method of claim 1, wherein said processing further comprises:

identifying data stored by the first database engine that is responsive to the database command; and

accessing said identified data, wherein said identifying and accessing are performed exclusively through the command layer of the alternate database engine, without interaction with the command layer of the first database engine.

24. (Previously Presented) The method of claim 1, wherein the alternate database engine executes only read-only database commands.

25. (Canceled)

26. (Previously Presented) An apparatus for processing a database command, comprising:

a processor; and

a memory in operative communication with the processor, the memory for storing a plurality of processing instructions for directing the processor to:

receive, from a user, a database command requiring data from a first database engine, the first database engine having a command layer for processing database commands; and

separately process the database command using a command layer of an alternate database engine without accessing the command layer of the first database engine.

27. (Previously Presented) A computer-readable medium encoded with processing instructions for implementing a method for processing a database command, performed by an alternate database engine, the method comprising:

receiving, from a user, a database command requiring data from a first database engine, the first database engine having a command layer for processing database commands; and

separately processing the database command using a command layer of an alternate database engine without accessing the command layer of the first database engine.

28. (Previously Presented) A method for implementing and using an alternate database engine in conjunction with an established database engine, the method comprising:

providing access to a first database engine to a plurality of users on a computing system, the first database engine having a command layer for processing database commands;

establishing an alternate database engine on the computing system;

receiving a database command from one of the plurality of users, the database command directed to data stored by the first database engine; and

separately processing the database command using the alternate database engine without accessing the command layer of the first database engine.

29. (Currently Amended) The method of claim ~~29~~ 28, wherein the computer system is one or more of: a local area network, a wide area network, an intranet, an extranet, a wireless network and-/or the Internet.

30. (Currently Amended) The method of claim ~~29~~ 28, wherein the first database engine stores the data in a first database file and the alternate database engine stores data in a second database file.

31. (Currently Amended) The method of claim ~~29~~ 28, wherein the database command is compatible with one or more of: a Structured Query Language format, a Javascript Database Connectivity protocol and/or an Open-Database Connectivity protocol.

32. (Original) The method of claim 28, wherein the database command is a query.

33. (Original) The method of claim 32, said processing the database command further comprising:

evaluating the query.

34. (Previously Presented) The method of claim 33, said evaluating further comprising:

evaluating the query against system usage.

35. (Previously Presented) The method of claim 34, said evaluating further comprising:

evaluating the query based on one or more of : a parameter of the query, a number of relational databases to be accessed for the query, a size of a data field to be searched for the query, an availability of resources of a system maintaining the alternate database engine, an availability of resources of a system maintaining the first database engine, a number of relational database tables to be employed for the query, a limitation imposed on a size of a query result set, a number of columns of data to be returned in a query result set, a cost of a similar stored query and/or a number of function calls for the query.

36. (Original) The method of claim 34, further comprising:

submitting the query to the alternate database engine with a limit on a number of returns responsive to the query, based on said evaluating.

37. (Original) The method of claim 34, further comprising

editing the query, based on said evaluating.

38. (Original) The method of claim 34, further comprising:

rejecting the query, based on said evaluating.

39. (Previously Presented) The method of claim 33, wherein said evaluating comprises:

determining, prior to said processing, whether the database command requires accessing the data of the first database engine, and if not, accessing only data stored by the alternate database engine.

40. (Original) The method of claim 39, said determining further comprising:

translating the query to a native format of the alternate database engine.

41. (Original) The method of claim 33, said evaluating further comprising:

determining whether the query requires accessing temporally sensitive data, and if so, accessing a transaction log of the first database engine.

42. (Original) The method of claim 32, further comprising:

generating a result of the query

43. (Currently Amended) The method of claim 42, further comprising:

transmitting the result to [the] one of the plurality of users submitting the database command.

44. (Original) The method of claim 42, wherein said transmitting further comprises:

transmitting the result in a format of the first database engine.

45. (Original) The method of claim 28, ~~further comprising~~ further comprising:

storing second data in a database file maintained by the alternate database engine.

46. (Original) The method of claim 45, said processing further comprising:

determining whether the database command requires at least a portion of said second data, and if so, identifying said portion responsive to the database command.

47. (Previously Presented) The method of claim 28, further comprising:

receiving new data to be provided to the plurality of users; and

storing said new data in a database file maintained by the first database engine.

48. (Original) The method of claim 28, said processing further comprising:

translating the database command to a native format of the alternate database engine.

49. (Original) The method of claim 28, wherein said processing further comprises:

identifying data stored by the first database engine that is responsive to the database command; and

accessing said identified data, wherein said identifying and accessing are performed exclusively through a command layer of the alternate database engine, without interaction with the command layer of the first database engine.

50. (Previously Presented) The method of claim 28, wherein the alternate database engine executes only read-only database commands.

51. (Canceled)

52. (Previously Presented) An apparatus for implementing and using an alternate database engine in conjunction with an established database engine, comprising:

a processor; and

a memory in operative communication with the processor, the memory for storing a plurality of processing instructions directing the processor to:

provide access to a first database engine to a plurality of users on a computing system, the first database engine having a command layer for processing database commands;

establish an alternate database engine on the computing system;

receive a database command from one of the plurality of users, the database command directed to data stored by the first database engine; and

separately process the database command using the alternate database engine without accessing the command layer of the first database engine.

53. (Previously Presented) A computer-readable medium encoded with processing instructions for performing a method of implementing and using an alternate database engine in conjunction with an established database engine, the method comprising:

providing access to a first database engine to a plurality of users on a computing system, the first database engine having a command layer for processing database commands;

establishing an alternate database engine on the computing system;

receiving a database command from one of the plurality of users, the database command directed to data stored by the first database engine; and

separately processing the database command using the alternate database engine without accessing the command layer of the first database engine.

54. (Previously Presented) A method for implementing and using an alternate database engine in conjunction with an established database engine, the method comprising:

providing access to a first database engine to a plurality of users on a computing system, the first database engine having a command layer for processing database commands, the first database engine further maintaining a first database file including first data;

establishing an alternate database engine on the computing system, the alternate database engine maintaining a second database file including second data;

receiving a database command from one of the plurality of users, the database command directed to one or more of said first and second data; and/or

separately processing the database command using the alternate database engine without accessing the command layer of the first database engine.

55. (Previously Presented) The method of claim 54, wherein the computer system is one or more of: a local area network, a wide area network, an intranet, an extranet, a wireless network and/or the Internet.

56. (Original) The method of claim 54, wherein the first database engine stores the data in a first database file and the alternate database engine stores data in a second database file.

57. (Previously Presented) The method of claim 54, wherein the database command is compatible with one or more of: a Structured Query Language format, a Javascript Database Connectivity protocol and/or an Open-Database Connectivity protocol.

58. (Original) The method of claim 54, wherein the database command is a query.

59. (Original) The method of claim 58, said processing the database command further comprising:

evaluating the query.

60. (Previously Presented) The method of claim 59, said evaluating further comprising:

evaluating the query against system usage.

61. (Previously Presented) The method of claim 60, said evaluating further comprising:

evaluating the query based on one or more of : a parameter of the query, a number of relational databases to be accessed for the query, a size of a data field to be searched for the query, an availability of resources of a system maintaining the alternate database engine, an availability of resources of a system maintaining the first database engine, a number of relational database tables to be employed for the query, a limitation imposed on a size of a query result set,

a number of columns of data to be returned in a query result set, a cost of a similar stored query and/or a number of function calls for the query.

62. (Original) The method of claim 60, further comprising:

submitting the query to the alternate database engine with a limit on a number of returns responsive to the query, based on said evaluating.

63. (Original) The method of claim 60, further comprising

editing the query, based on said evaluating.

64. (Original) The method of claim 60, further comprising:

rejecting the query, based on said evaluating.

65. (Previously Presented) The method of claim 59, wherein said evaluating comprises:

determining, prior to said processing, whether the database command requires accessing the first data, and if not, accessing only said second data using the alternate database engine.

66. (Original) The method of claim 65, said determining further comprising:

translating the query to a native format of the alternate database engine.

67. (Original) The method of claim 59, said evaluating further comprising:

determining whether the query requires accessing the temporally sensitive data of said first data, and if so, accessing a transaction log of the first database engine.

68. (Original) The method of claim 58, further comprising:

generating a result of the query

69. (Previously Presented) The method of claim 68, further comprising:

transmitting the result to one of the plurality of users submitting the database command.

70. (Original) The method of claim 69, wherein said transmitting further comprises:

transmitting the result in a format of the first database engine.

71. (Original) The method of claim 54, further comprising:

receiving new data to be provided to the plurality of users; and

storing said new data in the first database file.

72. (Original) The method of claim 54, further comprising:

receiving new data to be provided to the plurality of users; and

storing said new data in the second database file.

73. (Original) The method of claim 54, said processing further comprising:

translating the database command to a native format of the alternate database engine.

74. (Original) The method of claim 54, wherein said processing further comprises:

identifying data stored by the first database engine that is responsive to the database command; and

accessing said identified data, wherein said identifying and accessing are performed exclusively through a command layer of the alternate database engine, without interaction with the command layer of the first database engine.

75. (Previously Presented) The method of claim 54, wherein the alternate database engine executes only read-only database commands.

76. (Canceled)

77. (Previously Presented) An apparatus for implementing and using an alternate database engine in conjunction with an established database engine, comprising:

a processor; and

a memory in operative communication with the processor, the memory for storing a plurality of processing instructions directing the processor to:

provide access to a first database engine to a plurality of users on a computing system, the first database engine having a command layer for processing database commands, the first database engine further maintaining a first database file including first data;

establish an alternate database engine on the computing system, the alternate database engine maintaining a second database file including second data;

receive a database command from one of the plurality of users, the database command directed to one or more of said first and second data; and/or

separately process the database command using the alternate database engine without accessing the command layer of the first database engine.

78. (Previously Presented) A computer-readable medium encoded with processing instructions for performing a method of implementing and using an alternate database engine in conjunction with an established database engine, the method comprising:

providing access to a first database engine to a plurality of users on a computing system, the first database engine having a command layer for processing database commands, the first database engine further maintaining a first database file including first data;

establishing an alternate database engine on the computing system, the alternate database engine maintaining a second database file including second data;

receiving a database command from one of the plurality of users, the database command directed to one or more of said first and second data; and/or

separately processing the database command using the alternate database engine without accessing the command layer of the first database engine.

79. (Previously Presented) A method for requesting data, performed by a user on a computing system, the method comprising:

transmitting a database command directed to data stored by a first database engine, the first database engine having a command layer for processing database commands; and

receiving a result of the database command from an alternate database engine,

wherein the database command is separately processed by the alternate database engine without accessing the command layer of the first database engine.

80. (Previously Presented) A method for requesting data, performed by a user on a computing system, the method comprising:

transmitting a database command directed to data stored by a first database engine and an alternate database engine, the first database engine having a command layer for processing database commands; and

receiving a result of the database command from an alternate database engine,

wherein the database command is separately processed by the alternate database engine without accessing the command layer of the first database engine.

81. (Previously Presented) A method for processing a database command, performed by an alternate read-only database engine, the method comprising:

receiving, from a user, a read-only database command requiring data from a first database engine, the first database engine having a command layer for processing database commands; and

separately processing the database command using a command layer of an alternate database engine without accessing the command layer of the first database engine and without executing any write commands and read-write commands.

82. (Currently Amended) A method for implementing and using an alternate database engine in conjunction with an established database engine, the method comprising:

providing access to a first database engine to a plurality of users on a computing system, the first database engine having a command layer for processing database commands;

establishing an alternate database engine on the computing system;

receiving a read-only database command from one of the plurality of users, the read-only database command directed to data stored by the first database engine; and

separately processing the database command using the alternate database engine without accessing the command layer of the first database engine, wherein the alternate database engine executes read-only database commands.

83. (Previously Presented) A method for implementing and using an alternate database engine in conjunction with an established database engine, the method comprising:

providing access to a first database engine to a plurality of users on a computing system, the first database engine having a command layer for processing database commands, the first database engine further maintaining a first database file including first data;

establishing an alternate database engine on the computing system, the alternate database engine maintaining a second database file including second data;

receiving a read-only database command from one of the plurality of users, the read-only database command directed to one or more of said first and second data; and/or

separately processing the database command using the alternate database engine without accessing the command layer of the first database engine, wherein the alternate database engine executes only read-write database commands.

84. (Currently Amended) A method for processing a database command, performed by an alternate database engine, the method comprising:

receiving, from a user, a database command directed to a first database engine, the first database engine having a command layer for processing database commands;

separately processing the database command using a command layer of the ~~alternative~~ alternate database engine without accessing the command layer of the first database engine, said processing further comprising:

evaluating the database command to determine system usage of the query at the database engine, prior to execution of the database command, said evaluating based on one or more of: a parameter of the query, a number of relational databases for the database command, a size of a data field to be searched for the database command, an availability of resources of the database engine, a number of relational database tables to be employed for the database command, a limitation imposed on a size of a query result set, a number of columns of data to be returned in a query result set, a cost of a similar stored database command and/or a number of function calls for the database command;

determining a threshold value for system usage of the alternate database engine, wherein the threshold value is based on one or more of: estimated processor usage, estimated memory usage, input/output resource usage and/or disk resource usage of the alternate database engine;

if the system usage surpasses a threshold value, performing one or more of the following: submitting the database command to the alternate database engine with a limit on a number of returns responsive to the database command, editing the database command, and/or rejecting the database command;

determining whether the database command requires accessing data maintained by the first database engine, and if not, accessing second data stored only by the alternate database engine;

determining whether the database command requires accessing temporally sensitive data, and if so, accessing a transaction log of the first database engine;

translating the database command to a native format of the alternate database engine;

generating a result of the database command; and

transmitting the result to the user in a format of the first database engine.

85. (Previously Presented) The apparatus of claim 26, further comprising:

receiving new data to be provided responsive to database commands; and

storing said new data in a database file maintained by the alternate database engine.

86. (Previously Presented) The apparatus of claim 26, further comprising:

receiving new data to be provided responsive to database commands; and

storing said new data in a database file maintained by the first database engine.

87. (Previously Presented) The apparatus of claim 26, said processing further comprising:

translating the database command to a native format of the alternate database engine.

88. (Previously Presented) The apparatus of claim 26, wherein said processing further comprises:

identifying data stored by the first database engine that is responsive to the database command; and

accessing said identified data, wherein said identifying and accessing are performed exclusively through the command layer of the alternate database engine, without interaction with the command layer of the first database engine.

89. (Previously Presented) The apparatus of claim 26, wherein the alternate database engine executes only read-only database commands.

90. (Previously Presented) A system for processing a database command, comprising:

receiving, from a user, a database command requiring data from a first database engine, the first database engine having a command layer for processing database commands; and

separately processing the database command using a command layer of an alternate database engine without accessing the command layer of the first database engine.

91. (Previously Presented) The system of claim 90, wherein the first database engine stores the data in a first database file.

92. (Previously Presented) The system of claim 90, wherein the alternate database engine stores second data in a second database file.

93. (Previously Presented) The system of claim 90, wherein the database command is compatible with one or more of: a Structured Query Language format, a Javascript Database Connectivity (JDBC) protocol and/or an Open-Database Connectivity protocol.

94. (Previously Presented) The system of claim 90, wherein the database command is a query.

95. (Previously Presented) The system of claim 90, further comprising:

storing second data in a database file maintained by the alternate database engine.

96. (Previously Presented) The system of claim 90, said processing further comprising:

determining whether the database command requires at least a portion of said second data, and if so, identifying said portion responsive to the database command.

97. (Previously Presented) The system of claim 90, further comprising:

receiving new data to be provided responsive to database commands; and

storing said new data in a database file maintained by the alternate database engine.

98. (Previously Presented) The system of claim 90, further comprising:

receiving new data to be provided responsive to database commands; and

storing said new data in a database file maintained by the first database engine.

99. (Previously Presented) The system of claim 90, said processing further comprising: translating the database command to a native format of the alternate database engine.

100. (Previously Presented) The system of claim 90, wherein said processing further comprises:

identifying data stored by the first database engine that is responsive to the database command; and

accessing said identified data, wherein said identifying and accessing are performed exclusively through the command layer of the alternate database engine, without interaction with the command layer of the first database engine.

101. (Previously Presented) The system of claim 90, wherein the alternate database engine executes only read-only database commands.

102. (Previously Presented) The method of claim 27, said processing the database command further comprising:

evaluating the query.

103. (Previously Presented) The method of claim 102, said evaluating further comprising:

evaluating the query against system usage.

104. (Previously Presented) The method of claim 103, said evaluating further comprising:

evaluating the query based on one or more of: a parameter of the query, a number of relational databases to be accessed for the query, a size of a data field to be searched for the query, an availability of resources of a system maintaining the alternate database engine, an availability of resources of a system maintaining the first database engine, a number of relational database tables to be employed for the query, a limitation imposed on a size of a query result set, a number of columns of data to be returned in a query result set, a cost of a similar stored query and/or a number of function calls for the query.

105. (Previously Presented) The method of claim 103, further comprising:

submitting the query to the alternate database engine with a limit on a number of returns responsive to the query, based on said evaluating.

106. (Previously Presented) The method of claim 103, further comprising

editing the query, based on said evaluating.

107. (Previously Presented) The method of claim 103, further comprising:

rejecting the query, based on said evaluating.

108. (Previously Presented) The method of claim 102, wherein said evaluating comprises:

determining, prior to said processing, whether the database command requires accessing the first database engine, and if not, accessing data stored only by the alternate database engine.

109. (Previously Presented) The method of claim 108, said determining further comprising:

translating the query to a native format of the alternate database engine.

110. (Previously Presented) The method of claim 102, said evaluating further comprising:

determining whether the query requires accessing temporally sensitive data, and if so, accessing a transaction log of the first database engine.

111. (Previously Presented) The method of claim 27, further comprising:

generating a result of the query

112. (Previously Presented) The method of claim 111, further comprising:

transmitting the result to the one of the plurality of users submitting the database command.

113. (Previously Presented) The method of claim 112, wherein said transmitting further comprises:

transmitting the result in a format of the first database engine.

114. (Previously Presented) The method of claim 27, further comprising:

storing second data in a database file maintained by the alternate database engine.

115. (Previously Presented) The method of claim 114, said processing further comprising:

determining whether the database command requires at least a portion of said second data, and if so, identifying said portion responsive to the database command.

116. (Previously Presented) The apparatus of claim 52, further comprising:

receiving new data to be provided to the plurality of users; and

storing said new data in a database file maintained by the first database engine.

117. (Previously Presented) The apparatus of claim 52, said processing further comprising:

translating the database command to a native format of the alternate database engine.

118. (Previously Presented) The apparatus of claim 52, wherein said processing further comprises:

identifying data stored by the first database engine that is responsive to the database command; and

accessing said identified data, wherein said identifying and accessing are performed exclusively through a command layer of the alternate database engine, without interaction with the command layer of the first database engine.

119. (Previously Presented) The apparatus of claim 52, wherein the alternate database engine executes only read-only database commands.

120. (Previously Presented) The method of claim 53, said processing the database command further comprising:

evaluating the query.

121. (Previously Presented) The method of claim 120, said evaluating further comprising:

evaluating the query against system usage.

122. (Previously Presented) The method of claim 121, said evaluating further comprising:

evaluating the query based on one or more of: a parameter of the query, a number of relational databases to be accessed for the query, a size of a data field to be searched for the query, an availability of resources of a system maintaining the alternate database engine, an availability of resources of a system maintaining the first database engine, a number of relational database tables to be employed for the query, a limitation imposed on a size of a query result set, a number of columns of data to be returned in a query result set, a cost of a similar stored query and/or a number of function calls for the query.

123. (Previously Presented) The method of claim 121, further comprising:

submitting the query to the alternate database engine with a limit on a number of returns responsive to the query, based on said evaluating.

124. (Previously Presented) The method of claim 121, further comprising:

editing the query, based on said evaluating.

125. (Previously Presented) The method of claim 121, further comprising:

rejecting the query, based on said evaluating.

126. (Previously Presented) The method of claim 120, wherein said evaluating comprises:

determining, prior to said processing, whether the database command requires accessing the first database engine, and if not, accessing data stored only by the alternate database engine.

127. (Previously Presented) The method of claim 126, said determining further comprising:

translating the query to a native format of the alternate database engine.

128. (Previously Presented) The method of claim 120, said evaluating further comprising:

determining whether the query requires accessing temporally sensitive data, and if so, accessing a transaction log of the first database engine.

129. (Previously Presented) The method of claim 53, further comprising:

generating a result of the query

130. (Previously Presented) The method of claim 129, further comprising:

transmitting the result to one of the plurality of users submitting the database command.

131. (Previously Presented) The method of claim 130, wherein said transmitting further comprises:

transmitting the result in a format of the first database engine.

132. (Previously Presented) The method of claim 53, further comprising:

storing second data in a database file maintained by the alternate database engine.

133. (Previously Presented) The method of claim 132, said processing further comprising:

determining whether the database command requires at least a portion of said second data, and if so, identifying said portion responsive to the database command.

134. (Previously Presented) The method of claim 129, further comprising:

receiving new data to be provided to the plurality of users; and

storing said new data in a database file maintained by the first database engine.

135. (Previously Presented) The method of claim 53, said processing further comprising:

translating the database command to a native format of the alternate database engine.

136. (Previously Presented) The method of claim 53, wherein said processing further comprises:

identifying data stored by the first database engine that is responsive to the database command; and

accessing said identified data, wherein said identifying and accessing are performed exclusively through a command layer of the alternate database engine, without interaction with the command layer of the first database engine.

137. (Previously Presented) The method of claim 53, wherein the alternate database engine executes only read-only database commands.

138. (Previously Presented) A system for implementing and using an alternate database engine in conjunction with an established database engine, comprising:

providing access to a first database engine to a plurality of users on a computing system, the first database engine having a command layer for processing database commands;

establishing an alternate database engine on the computing system;

receiving a database command from one of the plurality of users, the database command directed to data stored by the first database engine; and

separately processing the database command using the alternate database engine without accessing the command layer of the first database engine.

139. (Previously Presented) The system of claim 138, wherein the computer system is one or more of: a local area network, a wide area network, an intranet, an extranet, a wireless network and-/or the Internet.

140. (Previously Presented) The system of claim 138, wherein the first database engine stores the data in a first database file and the alternate database engine stores data in a second database file.

141. (Previously Presented) The system of claim 138, wherein the database command is compatible with one or more of: a Structured Query Language format, a Javascript Database Connectivity protocol and/or an Open-Database Connectivity protocol.

142. (Previously Presented) The system of claim 138, wherein the database command is a query.

143. (Previously Presented) The system of claim 138, further comprising:

storing second data in a database file maintained by the alternate database engine.

144. (Previously Presented) The system of claim 143, said processing further comprising:

determining whether the database command requires at least a portion of said second data, and if so, identifying said portion responsive to the database command.

145. (Previously Presented) The system of claim 138, further comprising:

receiving new data to be provided to the plurality of users; and

storing said new data in a database file maintained by the first database engine.

146. (Previously Presented) The system of claim 138, said processing further comprising:

translating the database command to a native format of the alternate database engine.

147. (Previously Presented) The system of claim 138, wherein said processing further comprises:

identifying data stored by the first database engine that is responsive to the database command; and

accessing said identified data, wherein said identifying and accessing are performed exclusively through a command layer of the alternate database engine, without interaction with the command layer of the first database engine.

148. (Previously Presented) The system of claim 138, wherein the alternate database engine executes only read-only database commands.

149. (Previously Presented) The apparatus of claim 77, said processing further comprising:

translating the database command to a native format of the alternate database engine.

150. (Previously Presented) The apparatus of claim 77, wherein said processing further comprises:

identifying data stored by the first database engine that is responsive to the database command; and

accessing said identified data, wherein said identifying and accessing are performed exclusively through a command layer of the alternate database engine, without interaction with the command layer of the first database engine.

151. (Previously Presented) The apparatus of claim 77, wherein the alternate database engine executes only read-only database commands.

152. (Previously Presented) A system for implementing and using an alternate database engine in conjunction with an established database engine, comprising:

providing access to a first database engine to a plurality of users on a computing system, the first database engine having a command layer for processing database commands, the first database engine further maintaining a first database file including first data;

establishing an alternate database engine on the computing system, the alternate database engine maintaining a second database file including second data;

receiving a database command from one of the plurality of users, the database command directed to one or more of said first and second data; and/or

separately processing the database command using the alternate database engine without accessing the command layer of the first database engine.

153. (Previously Presented) The system of claim 152, wherein the computer system is one or more of: a local area network, a wide area network, an intranet, an extranet, a wireless network and/or the Internet.

154. (Previously Presented) The system of claim 152, wherein the first database engine stores the data in a first database file and the alternate database engine stores data in a second database file.

155. (Previously Presented) The system of claim 152, wherein the database command is compatible with one or more of: a Structured Query Language format, a Javascript Database Connectivity protocol and/or an Open-Database Connectivity protocol.

156. (Previously Presented) The system of claim 152, wherein the database command is a query.

157. (Previously Presented) The system of claim 152, further comprising:

receiving new data to be provided to the plurality of users; and

storing said new data in the first database file.

158. (Previously Presented) The system of claim 152, further comprising:

receiving new data to be provided to the plurality of users; and

storing said new data in the second database file.

159. (Previously Presented) The system of claim 152, said processing further comprising:

translating the database command to a native format of the alternate database engine.

160. (Previously Presented) The system of claim 152, wherein said processing further comprises:

identifying data stored by the first database engine that is responsive to the database command; and

accessing said identified data, wherein said identifying and accessing are performed exclusively through a command layer of the alternate database engine, without interaction with the command layer of the first database engine.

161. (Previously Presented) The method of claim 78, wherein the computer system is one or more of: a local area network, a wide area network, an intranet, an extranet, a wireless network and/or the Internet.

162. (Previously Presented) The method of claim 78, wherein the first database engine stores the data in a first database file and the alternate database engine stores data in a second database file.

163. (Previously Presented) The method of claim 78, further comprising receiving new data to be provided to the plurality of users; and

storing said new data in the first database file.

164. (Previously Presented) The method of claim 78, further comprising:

receiving new data to be provided to the plurality of users; and

storing said new data in the second database file.

165. (Previously Presented) The method of claim 78, said processing further comprising:

translating the database command to a native format of the alternate database engine.

166. (Previously Presented) The method of claim 78, wherein said processing further comprises:

identifying data stored by the first database engine that is responsive to the database command; and

accessing said identified data, wherein said identifying and accessing are performed exclusively through a command layer of the alternate database engine, without interaction with the command layer of the first database engine.

167. (Previously Presented) The method of claim 78, wherein the alternate database engine executes only read-only database commands.